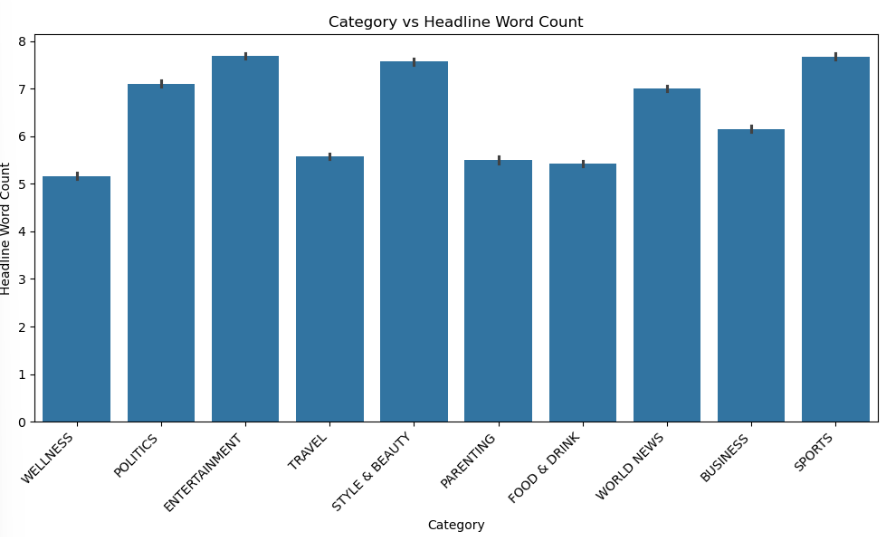
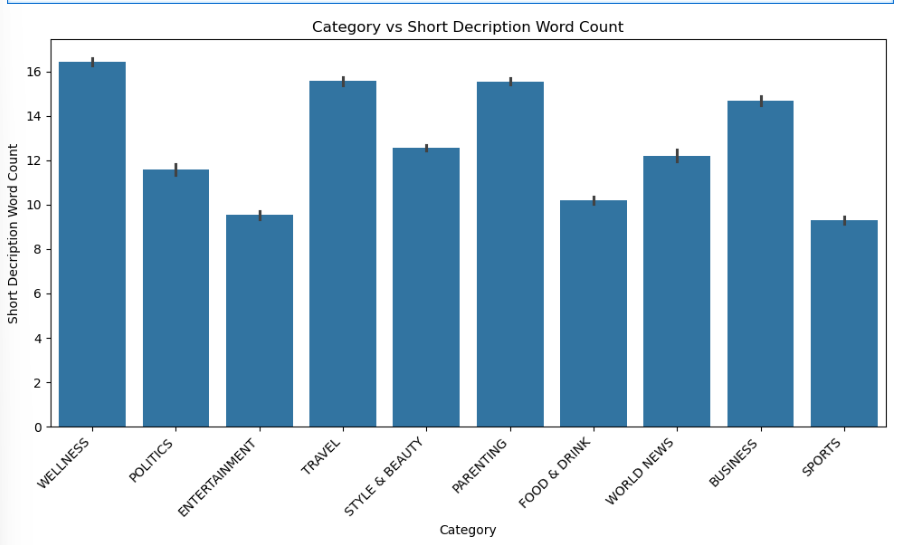
# Final Report on Part B News Article Classification

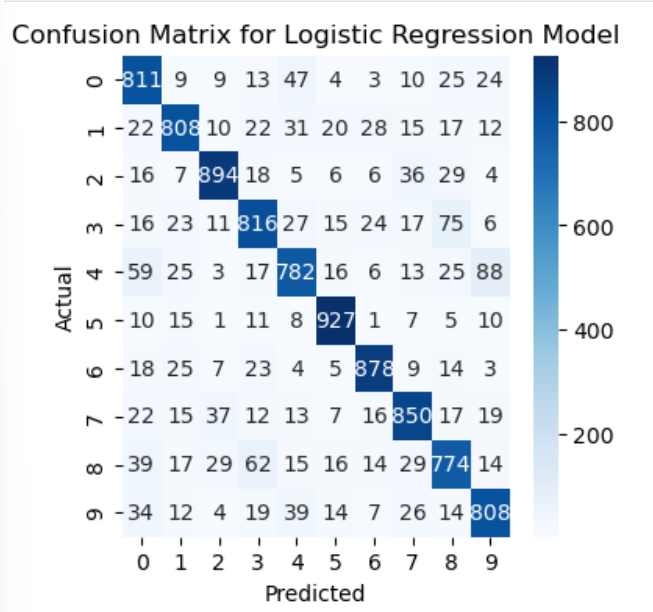
* Step 1: Import Libraries
* Initially imported all the required Libraries in Jupyter Notebook.
* Step 2: Data Collection and Preprocessing
* Then loaded dataset and created dataframe.
* When analyzed the dataset for missing values, found that there are missing values in the keywords feature. Replaced them with frequently occurring value.
* Then converted text columns to lowercase. And created functions for removing punctuation and special characters. Also created function to tokenize text columns.
* Removed "not" word from stopword list. As it might have impact on negative news articles.
* Have created function to remove stopwords and extra spaces from text columns. Also created function to remove hyphen from only keyword column.
* Then created function to perform lemmatization. Because meaning of the words in the news article is crucial in this scenario.
* Finally joined the cleaned text into single string.
* Have applied all the above functions to the text columns in news df.
* Step 3: Feature Extraction
* Created textual features like headline\_word\_count, short\_decription\_word\_count, headline\_char\_count, short\_description\_char\_count, headline\_avg\_word\_length, short\_description\_avg\_word\_length using headline, short\_description and keywords column.
* Combined the headline, short\_description and keywords columns into a single dataframe called combined text column.
* On this combined text column, performed TF-IDF vectorisation. Because importance of words in news article helps in analysing them in a better way. So, TF-IDF is a better vectorisation model for this dataset.
* Step 4: Performing Exploratory Data Analysis (EDA)
* The dataset is a balanced dataset as the number of records for each category are same.
* Created bar plot for category vs headline\_word\_count feature.



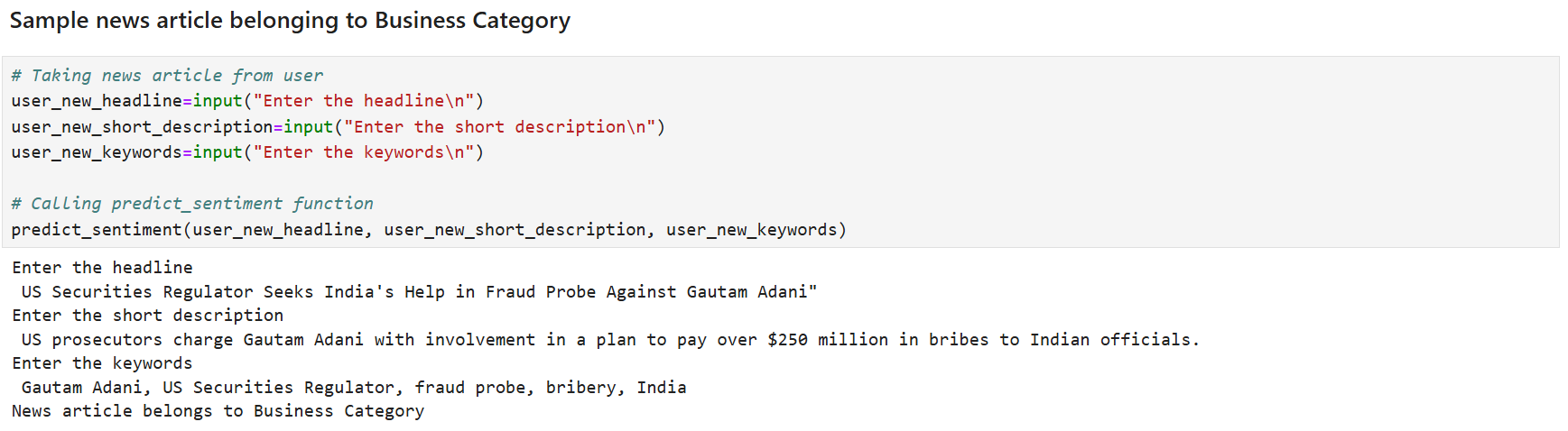
* Entertainment category has highest value for headline word count feature.
* Style & Beauty category and Sports category have next highest value for headline word count feature.
* Wellness category has lowest value for headline word count feature.
* Travel category, Parenting category and Food and Drink category have next lowest value for headline word count feature.
* Created bar plot for category vs short\_decription\_word\_count feature.



* Wellness category has the highest value for short description word count feature.
* Travel category and Parent category have next highest value for short description word count feature.
* Sports category has lowest value for short description word count feature.
* Entertainment category and Food and Drink category have next lowest value for short description word count feature.
* Step 5: Model Development
* Encoded all the categories to numerical format using label encoder.
* Combined TF-IDF features with numerical features to create final dataset.
* Have split the dataset into 80% as train set and 20% as test set.
* With Default values, Logistic Regression Model is performing poorly on test data with an accuracy of 23.88%. Performing hyperparameter tuning to get better accuracy.
* After hyperparameter tuning for Logistic Regression model, accuracy has increased from 23.88% to 82.86%. We can consider this model during model evaluation as the accuracy value is highest among all other models. See below for reference.
* With Default values, Random Forest Model is performing well on test data with an accuracy of 75.77%. Performing hyperparameter tuning to get better accuracy.
* After hyperparameter tuning for Random Forest model, accuracy has decreased from 75.77% to 69.33%. Since the accuracy value is less than logistic regression model, not considering this model during model evaluation.
* With Default values, SVM Model is performing poorly on test data with an accuracy of 23.88%. So, not considering this model for model evaluation.
* Step 6: Model Evaluation
* Among all the models, Logistic regression is giving highest accuracy. So, considering this model for evaluating the model performance.
* Accuracy is improved from 82.86% to 83.48% on unseen data when hyperparameter tuned values are considered. So, we can consider this model for predicting categories of a news article.
* We have high precision and recall for all the categories. This indicates that the model is able to correctly predict the categories for the given news article.
* Step 7: Visualisation
* Generated confusion matrix on the predicted values of logistic regression model.



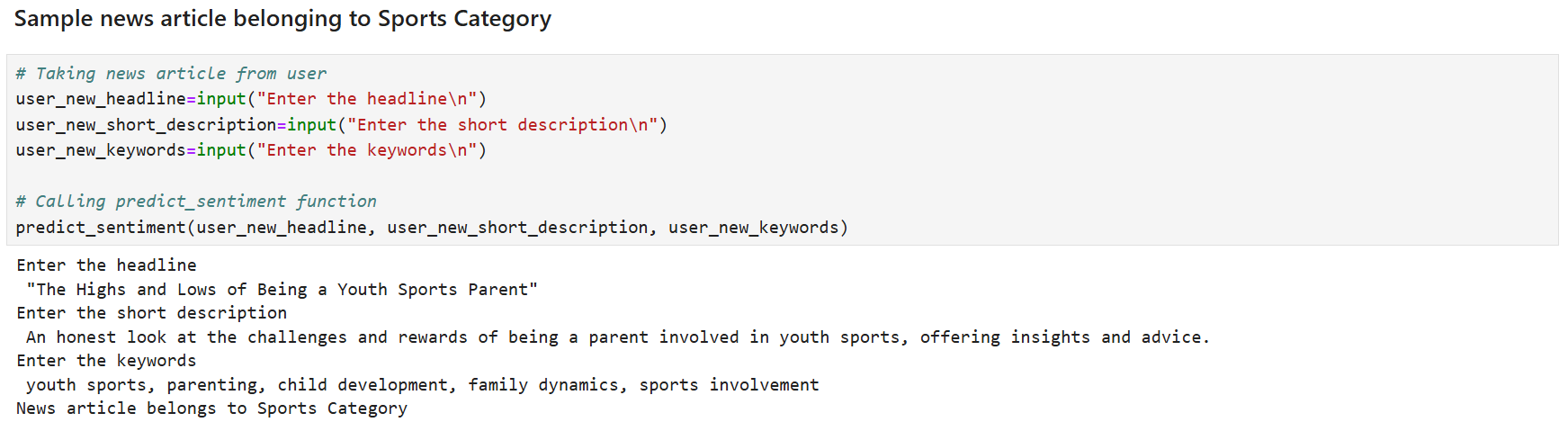
* All Diagonal elements of the confusion matrix have high values indicating that the model is correctly classifying instances for all the classes.
* We have low values in all off diagonal matrix indicating that the misclassification of the classes is less by the model.
* Hence, we can prefer Logistic Regression Model for predicting categories for the unseen or future news article.
* Step 8: Prediction on user input
* Created pre-processing text function to preprocess future news article given by user.
* Created predict sentiment function to return the category of the news article.
* When user inputs new or unseen news article, then respective category name will be displayed.
* Example 1: Business Category



* Example 2: World News Category



* Example 3: Sports Category



* Final Insights:
* 1. News Article dataset is a large dataset with 50k records of news records.
* 2. Have trained the dataset with Logistic Regression, Random Forest and SVM algorithms.
* 3. Among these algorithms, Logistic Regression Algorithm works better as it has accuracy value of 82.86 percent.
* 4. This indicates that the model is able to predict the new or unseen data correctly 82 times out of 100 times.
* 5. From confusion matrix, we observe a high value for all diagonal elements. This implies that the model is correctly predicting category as to which the news article belongs to.
* 6. So, we can completely rely on this model for future or unseen news article to know its category.